Remarks

This reply is in response to the Office communication dated April 1, 2005. Unless otherwise indicated, page references are to that communication.

Claim 1 has been amended to recite that the single hardware instruction referenced there is capable of processing a plurality of characters "in a single invocation of the instruction". This accurately characterizes the Translate Two to One (TRTO) instruction referenced in the specification, as well as in publicly available documentation such as the z/Architecture Principles of Operation, SA22-7832-00 (Dec. 2000), an excerpt of which (pages 3 to 4 and 7-152 to 7-157) accompanies this amendment as an attachment. As described on page 7-153 of that publication, on execution of a TRTO instruction specifying first and second operands:

> The characters of the second operand are used as arguments to select function characters from a translation table designated by the address in general register 1. Each function character selected from the translation table is compared to a test character in general register 0, and, unless an equal comparison occurs, is placed at the first-operand location. The operation proceeds until a selected function character equal to the test character is encountered, the end of the second operand is reached, or a CPU-determined number of characters have been processed, whichever occurs first.

Claim 14 has been amended to recite that the sub-codepages "other than said highest-priority sub-codepage" comprise a first set of one or more higher-priority sub-codepages and a second set of one or more lower-priority sub-codepages. While this limitation is believed to have been implicit in the claim as previously presented, it is now made explicit.

Claims 1-13

Claims 1-13 stand rejected under 35 U.S.C. §§ 102(e) and 103(a) as being either anticipated by or unpatentable over U.S. Patent 6,204,782 to Gonzalez et al. ("Gonzalez").

Claim 1 as currently amended is directed to a method for converting a source string in which the characters are converted using a single hardware instruction capable of processing a plurality of characters in a single invocation of the instruction (emphasis added).

The Examiner contends, in his response to applicant's previous arguments, that Gonzalez "in fact does disclose the claimed limitation of using a single hardware instruction capable of processing a plurality of characters" (page 8, ¶ 8). As evidence of this, the Examiner points to the flow diagram of Fig. 7, as well as the "hardware implementation" shown in Fig. 10. The Examiner concludes from Fig. 7 that "[i]t is clear . . . that only a single hardware instruction is needed to start the process of converting the source string into the target string because the text processing automatically continues until the entire string is converted" (pages 8-9). Applicant respectfully disagrees.

The Examiner will note that his conclusion—that only a single hardware instruction is needed to start the process of converting the source string into the target string—does not quite match the assertion made elsewhere—namely, that Gonzalez discloses a single hardware instruction capable of processing a plurality of characters. The mere fact that a single hardware instruction is capable of starting a conversion process says nothing about whether that same instruction is capable of completing the conversion process. In fact, it would appear that Gonzalez requires a multiplicity of instruction invocations to complete even a single loop of the procedure shown in Fig. 7. He would require even more instruction invocations, of course, to perform the entire conversion process comprising multiple traversals of the loop shown in Fig. 7.

The Examiner refers to the "hardware implementation" shown in Fig. 10. However, that implementation is merely a general-purpose computer, with no special enhancements for performing the disclosed method. And while the patentee notes that the computer system "may be specially constructed for the required purposes" (col. 19, lines 56-57), he gives no examples of any such specially constructed system, much less an example using a particular machine instruction.

For the foregoing reasons, applicant respectfully submits that claim 1 as amended and the claims dependent thereon distinguish patentably over the art cited by the Examiner.

Claims 14-20

Claims 14-20 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Gonzalez.

Claims 14-20 as amended are directed to that feature of applicant's invention whereby the sub-codepages other than the highest-priority sub-codepage comprise a first set of one or more higher-priority sub-codepages and a second set of one or more lower-priority sub-codepages. If a character is found in a sub-codepage belonging to the first set of sub-codepages (i.e., the higher-priority sub-codepages), the conversion process is continued with that sub-codepage as the current sub-codepage. On the other hand, if the character is found in a sub-codepage belonging to the second set of sub-codepages, the character is converted using that sub-codepage and the conversion process is continued with the highest-priority sub-codepage as the current sub-codepage.

In rejecting these claims, the Examiner seems to argue that claim 14 as previously presented reads on Gonzalez's first approach of always reverting to the default codepage after a successful conversion, since the default codepage could be regarded as a constituting the only member of a first set of code pages.[1] It would have been obvious, the Examiner argues, "to have modified Gonzalez . . . to have created a high-priority sub-code page set instead of only a highest-priority sub-codepage so that a block of source text including two major sub-codepages could have translated both of the high-priority sub-codepages of the source text string according to the first approach as taught by Gonzalez in fig. 7 and col. 15 line 32 - col. 17 line 45" (page 6). Applicant respectfully disagrees.

[1] It is unclear to applicant whether the Examiner is saying that Gonzalez anticipates claim 14 (because the highest-priority sub-codepage is a set of one) or merely renders it obvious (because one could enlarge the set to more than one). Applicant has never argued, however, that a "set" requires more than one member, and the claims as currently worded (". . . set of one or more . . .") imply that one member suffices. In any event, applicant has amended claim 14 to specify that the sub-codepages other than the first sub-codepage comprise the first and second sets.

In describing his approaches, the patentee distinguishes between a bias toward "a preferred target encoding" (emphasis added) and attempting to minimize switching between target encodings (col. 6, lines 45-51). There is no notion here of having more than a single "preferred" target encoding for the purposes of minimizing switching between encodings. Rather, the presumption is that the first encoding differs in kind from all lower-priority encodings, since this is where one always goes after an unsuccessful conversion attempt. Indeed, the only suggestion in the record for having more than a single "preferred" encoding comes from applicant's own disclosure and not Gonzalez or any other cited art. This is clearly hindsight, however, and thus impermissible.

For the foregoing reasons, applicant respectfully submits that claims 14-20 distinguish patentably over the art cited by the Examiner.

Conclusion

Reconsideration of the application as amended is respectfully requested. It is hoped that upon such consideration the Examiner will hold all claims allowable and pass the case to issue at an early date. Such action is earnestly solicited.

Respectfully submitted,
JOACHIM M. BAUER

By

William A. Kinnaman, Jr.
Registration No. 27,650
Phone: (845) 433-1175
Fax: (845) 432-9601

WAK/wak